

An Optimized Classification Method for Multi-Classed Sequences

Chieh-Yuan Tsai, Chih-Jung Chen, and Wen-Jen Lee

Abstract—Multi-classes imply an additional difficulty for sequence classification problem, since the boundaries among these classes can be overlapped. To increase classification accuracy, this research divides sequences into a set of sequence subsets according to the class label of sequences. Then, a sequential pattern mining algorithm is applied for each sequence subset. The discovered sequential patterns will be used as the representation for classes. Next, the pairwise coupling method is used for every pair of sequence subsets and form a set of binary class datasets. For each binary sequence dataset, a Particle Swarm Optimization with Simulated Annealing algorithm based Binary Sequence classifier, named PSO-SA-BS classifier, is constructed. In the PSO-SA-BS classifier, the PSO-SA optimization algorithm is developed to update the weights in the classifier so that the classification accuracy of each classifier can be maximized. Finally, to aggregate the output from all PSO-SA-BA classifiers, the fuzzy preference relation between each pair of binary database is evaluated. According to the fuzzy preference relation, a class label with largest class score is assigned to the sequence. The experiments show that the performance of the proposed method is higher than other classical classification methods.

Index Terms—Multi-class classification problem, sequential pattern mining, particle swarm optimization, simulated annealing, fuzzy preference relation.

I. INTRODUCTION

The sequence classification is to assign a most probable class label to a given sequence by a generative model (or classifier). Actually, sequence classification problem can be found in many real-world applications such as protein function prediction, text classification, speech recognition, intrusion detection, and customer behavior prediction. For example, when sequence classification is applied to protein function prediction, user will understand the function of the specific protein from the complex and the large amount of protein data much easier than the typical method like X-ray diffraction or nuclear magnetic resonance (NMR) for protein function prediction. Additionally, when sequence classification technique is applied to customer behavior prediction, users may predict the purchase behavior of a new customer according to the classification model constructed

by the customer transaction record. Reference [1] employed support vector machine (SVM) to develop sequence classification model. Reference [2] employed hidden Markov model (HMM), a statistical model in which the system being modeled is assumed to be a Markov process with unobserved state, to generate a sequence classification model. Reference [3] extended the traditional hidden Markov model named “structural hidden Markov model” by partitioning the set of observation sequences into classes of equivalences to enhance the accuracy for the sequence classification model. Reference [4] integrated estimation, clustering, and classification into the traditional, three-step approach to make the result of sequence classification more reliable.

Recently, sequential pattern classification problems are considered as feature mining problem. That is, the sequential patterns are discovered by a set of training sequences and then use these patterns for classification. The feature mining algorithm uses the extracted patterns as features. The patterns are vectorized based on their matched sequences and then standard classification method algorithms such as Naïve Bayes or Winnow are applied to the vectorized sequences. Reference [5] proposed a feature mining technique called “scalable feature mining” which is the improved FeatureMine algorithm for sequential data to act as the preprocessor to select features for standard classification algorithm such as Winnow and Naïve Bayes. By adapting scalable and disk-based data mining algorithms, they are able to classify the sequences efficiently. Reference [6] used a probabilistic model based on hidden Markov models (HMMs) to address the sequence classification problem. In contrast to commonly-used likelihood-based learning methods such as the joint/conditional maximum likelihood estimator, they introduced a discriminative learning algorithm that focuses on class margin maximization.

Multiclass implies an additional difficulty for sequence classification problem, since the boundaries among these classes can be overlapped which causes a decrease of the performance. In the domain of sequence classification problems, most of previous studies focus on two class problem that are positive class and negative class. However, in real world, multi-class problem is everywhere around us.

Therefore, the major objective of this research is to improve the prediction accuracy when data are multi-classes sequences. In order to achieve the objective, we transform the original multi-class problem into binary subsets, which are easier to discriminate, via a pairwise coupling method. For each classifier, the particle swarm optimization (PSO) algorithm with simulated annealing (SA) algorithm to adjust and optimize the weights in the classifier. Finally, fuzzy preference relation is used to integrate the results of all

Manuscript received October 16, 2012; revised December 27, 2012.

Chieh-Yuan Tsai is with the Institute of Industrial Engineering and Management, Yuan-Ze University, Chungli City, Taoyuan County, 320 Taiwan. (e-mail: cytsai@saturn.yzu.edu.tw).

Chih-Jung Chen is with the Department of Industrial Engineering and Management at Yuan Ze University, Taiwan. (e-mail: s968910@mail.yzu.edu.tw).

Wen-Jen Lee is with the Institute of Industrial Engineering and Management, Yuan-Ze University, Taiwan (e-mail: s985445@mail.yzu.edu.tw).

sub-classifiers and to calculate the final classification result.

II. RESEARCH METHOD

A. Research Framework

The framework of the proposed sequence classification method is depicted in Fig. 1. A sequence database SD is divided into n sub-databases according to the class label which each sequence belongs to. Let the c th sub-database be presented as $SD^c = \{ \langle S_j, c \rangle \}$, where S_j represents the j th sequence and c represents the class for this sequence where $c = 1, 2, \dots, n$. There are four main steps in the proposed method. First, a sequential pattern mining algorithm, AprioriAll, is applied for SD^c where $c = 1, 2, \dots, n$. Then, the frequent sequential pattern set for class c , denoted as SP^c , is generated for $c = 1, 2, \dots, n$. Second, pairwise coupling approach is used to generate $n \times (n-1) / 2$ binary datasets where each binary dataset is represented as $\langle \{SD^a, SD^b\}, \{SP^a, SP^b\} \rangle$ where $a, b \in \{1, 2, \dots, n\}, a \neq b$. For example, if $n=3$, three binary datasets can be generated as $\langle \{SD^1, SD^2\}, \{SP^1, SP^2\} \rangle$, $\langle \{SD^1, SD^3\}, \{SP^1, SP^3\} \rangle$, $\langle \{SD^2, SD^3\}, \{SP^2, SP^3\} \rangle$. Third, a Particle Swarm Optimization with Simulated Annealing algorithm based Binary Sequence classifier, named PSO-SA-BS classifier, is constructed for each binary sequence dataset. In the PSO-SA-BS classifier, the similarity between two sequences is developed by a dynamic similarity computation method. Then, the PSO-SA optimization algorithm is developed to update the weights in the classifier so that the classification accuracy of each classifier can be maximized. Fourth, to aggregate the output from all BS classifiers, the fuzzy preference relation between each pair of binary database is evaluated. According to the fuzzy preference relation, a class label with largest class score is assigned to the sequence.

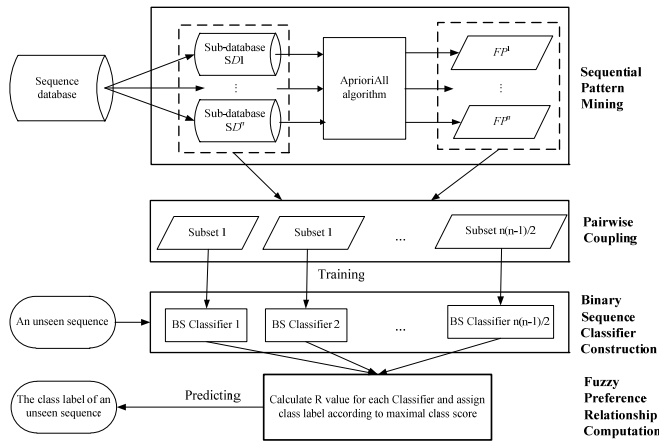


Fig. 1. Framework of the sequence classification method.

B. Sequential Pattern Mining

A sequence S is an ordered list of itemsets, denoted as $\langle s_1, s_2, \dots, s_n \rangle$, where s_i is an itemset. An itemset s_i is called an element of the sequence, denoted as $s_i = (i_1, i_2, i_3, \dots, i_m)$ where i_k is an item. A sequence $\langle a_1, a_2, \dots, a_n \rangle$ is contained in another sequence $\langle b_1, b_2, \dots, b_m \rangle$ if there exist integers

$i_1 < i_2 < \dots < i_n$ such that $a_1 \subseteq b_{i_1}, a_2 \subseteq b_{i_2}, \dots, a_n \subseteq b_{i_n}$. If sequence S contains sequence S' , and S' is called a subsequence of S . The support for a sequence is defined as the fraction of total customers who support this sequence. Given a transaction database SD , sequential pattern mining is to find the maximal sequences among all sequences that satisfy a certain user-specified minimum support. Each such maximal sequence represents a sequential pattern. An itemset with minimum support is called a large itemset or *Itemset*. The AprioriAll algorithm, which is proposed by [7], is used in this research.

C. The Pairwise Coupling Approach

In the domain of sequence classification problems, most of previous studies focus on two class problem that are positive class and negative class. However, in the real world, multi-class problem is everywhere around us. In order to generate a better classification result, a pairwise coupling approach is suggested to transform the original multi-class problem into binary class problem [8]. The pairwise coupling approach divides and composes the original dataset into $n(n-1)/2$ subsets with each pair of classes, where n stands for the total number of classes in the problem.

D. Particle Swarm Optimization with Simulated Annealing Algorithm based Binary Sequence Classifier

The flowchart of the proposed Particle Swarm Optimization with Simulated Annealing algorithm based Binary Sequence (PSO-SA-BS) classifier for each subset is shown in Fig. 2. To make the following discussion easier, we assume that a PSO-SA-BS classifier deals with sequences with class labels x and y .

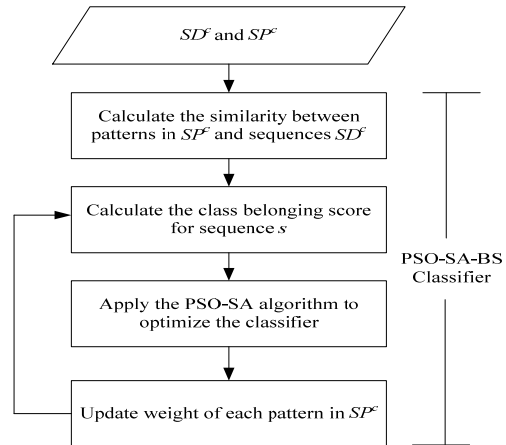


Fig. 2. The flowchart of BS classifier.

The dissimilarity between a sequence s in SD^c and a sequential pattern sp in SP^c is calculated based on the concept of dynamic programming. The minimal penalty cost between a sequence and a pattern is evaluated by sequential pattern matching, also called edit distance. Generally speaking, four basic edit operations are considered which are “substitution”, “insertion”, “deletion”, and “no change”. Besides, the penalty cost of each “substitution”, “insertion”, and “deletion” is set to be 1, and the penalty cost for taking each “no change” is equal to 0 at most traditional string matching

cases [9]. Then, the edit operation with minimal penalty cost is selected in each evaluation step. Based on the above concept, the penalty cost of essential edit operations is summarized as in (1):

$$Cost_{p,q} = \begin{cases} 1, & \text{if Insertion or Deletion} \\ 1, & \text{if Substitution} \\ 0, & \text{else No Change} \end{cases} \quad (1)$$

where p means the p th element in sequential pattern sp , q means the q th element in sequence s . The penalty cost for edit distance is the degree of dissimilarity between sp and s . Hence, the similarity between each edit operation is defined as 1 minus the penalty cost of each edit operation, which is defined as in (2):

$$Sim_{p,q} = 1 - Cost_{p,q} \quad (2)$$

The total similarity between the whole sequence and the whole sequential pattern, is the sum of all $Sim_{p,q}$ divided by the longest length between the sequence and the sequential pattern. This normalization operation is set to avoid the condition the longer sequences obtain higher similarity value with a specific pattern which might affects the fairness of similarity measure. The similarity between sequential pattern sp and sequence s is summarized as follow:

$$Sim(sp,s) = \frac{\sum_p \sum_q^{l_{sp}} Sim_{p,q}}{\max(l_s, l_p)} \quad (3)$$

where l_s denotes the length of sequence s and l_{sp} denotes the length of sequential pattern sp . After defining the similarity measure between sequential pattern sp and sequence s , the belonging score M_s^c that evaluates how strong a sequence s belongs to class c can be derived as:

$$M_s^c = \sum_{k=1}^{q_c} wp_k^c \times Sim(sp_k^c, s) \quad (4)$$

where sp_k^c is k th sequential pattern in SP^c , wp_k^c is the important weight of k th sequential pattern sp_k^c , and $Sim()$ is the similarity measure defined in (3). Note that each sequential pattern has an important weight wp_k^c which will be adjusted automatically by the PSO-SA algorithm. Therefore, when a sequence input to the PSO-SA-BS classifier, a class label will be return as:

$$PSO-SA-BS \text{ classifier}(s) = \arg \max_{c=\{x,y\}} (M_s^c) \quad (5)$$

As mentioned in (4), the important weight wp_k^c in each PSO-SA-BS classifier might dramatically affect the accuracy of the proposed classification method. Therefore, to obtain the best classification result, the set of weights wp_k^c are optimized using the proposed Particle Swarm Optimization with Simulated Annealing (PSO-SA) algorithm. In the PSO algorithm, each individual is considered as a particle that

presents a candidate solution to the optimization problem. The position of a particle is adjusted according to its own best experience and the best experience of other particles. This adjustment procedure is repeated until stopping criteria such as all particles converge to the best solution are reached. However, PSO algorithm tends to trap the current best solution into the local optimal solution in the searching space [10]. Therefore, the SA algorithm will be integrated with PSO algorithm in this research to overcome this problem.

In PSO-SA hybrid algorithm, the initial solution, which includes position and velocity of a particle, of each individual are randomly generated. Then, the positions of individual particle are adjusted according to its velocity which will be changed by previous best positions and the neighborhood best or the global best. Specifically, each particle is updated iteratively by:

$$V_i^{t+1} = wV_i^t + c_1 rand_1(P_i^t - X_i^t) + c_2 rand_2(P_g^t - X_i^t) \quad (6)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (7)$$

where $i = 1, 2, \dots, N_{swarm}$ is the index of each particle, N_{swarm} is the total number of particles, t is the iteration number, $rand_1$ and $rand_2$ are random numbers between 0 and 1. X_i^t is the current solution in iteration t . P_i^t is the best previous solution of the i th particle that is recorded, P_g^t is the best particle among the whole population. w is a positive constant which represent the weight of previous velocity, c_1 and c_2 are also positive constants which represent the weight of the stochastic acceleration terms that pull each particle toward P_i^t and P_g^t respectively. Hence, the values of c_1 and c_2 are often set to 2 according to early experiences [11].

To avoid that each particle moves too far, the velocity of each particle in (6) should be no greater than a maximum velocity [12], [13]. That is, if the new velocity derived from (6) is greater than the maximum velocity, the new velocity is set as maximum velocity. The maximum velocity can be calculated as follow:

$$V^{\max} = \gamma(X_{ub} - X_{lb}) \quad (8)$$

where X_{ub} and X_{lb} are the upper bound and lower bound in the searching space, and γ is a parameter to control the move distance of maximum velocity in the searching space.

In addition, this study applies two strategies to avoid particles become trapped in the local best solution if the objective function value for the global best solution in iteration t , $f(P_g^t)$, is no less than the objective function value for the global best solution in iteration $t-h$ $f(P_g^{t-h})$. The first strategy is to dynamically adjust the inertia weight (w) and maximum velocity (V^{\max}) [14]. The w and V^{\max} are adjusted as:

$$w_{t+1} = \alpha w_t \quad (9)$$

$$V_{t+1}^{\max} = \beta V_t^{\max} \quad (10)$$

where α and β , which are between 0 and 1, are the dynamic adjustment parameters for inertia weight and maximum velocity respectively. In this study, α and β are set to be 0.99 and 0.95 according to previous study [14].

The second strategy is to use simulated annealing (SA) algorithm to help the global best solution P_g^t in PSO algorithm escape its current local optimal space [10], [15]. The new possible solution $P_{g_new}^t$ can be generated by:

$$\begin{cases} P_{g_new}^t = P_g^t + R_2(X_{ub} - P_g^t), & \text{if } R_1 > 0.5 \\ P_{g_new}^t = P_g^t + R_2(P_g^t - X_{lb}), & \text{if } R_1 \leq 0.5 \end{cases} \quad (11)$$

where R_1 is a random number to decide the searching direction and R_2 is a random number to decide the and searching range respectively. X_{ub} and X_{lb} are the upper bound and lower bound in the searching space. If the new possible solution generated by (11) is greater than the global best solution, the global best solution is replaced by the new possible solution. However, if the new possible solution is worse than the global best solution, the Boltzmann probability will be used to determine whether the global best solution is replaced by the new possible solution or not. The Boltzmann probability can be shown as:

$$P = \left(\frac{-\Delta f}{kT_j} \right) \quad (12)$$

where k is Boltzmann's constant and set as 1 usually, Δf is a difference of the objective function values between two continuous iterations, and T_j is the temperature of j th iteration.

E. Fuzzy Preference Relations

After all PSO-SA-BS classifiers are built, the fuzzy preference relation will be applied to conform the result of every classifier. When a new sequence s is input to a classifier that deals with classes x and y , the classifier will compute the belonging score between sequence s and class x , denoted as M_x , and the belonging score between sequence s and class y , denoted as M_y . Based on the two belonging scores, the fuzzy preference relation between classes x and y under sequence s , can be defined as:

$$R(x, y) = \frac{M_x}{M_x + M_y} \quad (13)$$

where $R(x, y)$ is the maximum degree of similarity for all patterns that concludes class x . $R(x, y)$ will be ranged between 0 and 1 and have the relation $R(x, y) = 1 - R(y, x)$. Equation (13) will be conducted $C(C-1)$ times so that the fuzzy preference relations between all pairs of classes can be obtained. Notes that $R(1, 1), R(2, 2), \dots, R(C, C)$ are set as 1.

Based on these fuzzy preference relations, non-dominated class value for class c , ND_c , is extracted where ND_c is the degree that the new sequence does not belong to other classes. The non-dominated class value for class c , ND_c , can be evaluated as [16]:

$$ND_x = 1 - \sup_{y \in C} [R'(y, x)] \quad (14)$$

where the fuzzy strict preference relation between x and y , $R'(x, y)$, is defined as

$$R'(x, y) = \begin{cases} R(x, y) - R(y, x), & \text{when } R(x, y) > R(y, x) \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

The index of the maximal non-dominance value among all classes will be the output class for sequence s which is computed as:

$$\text{Class}(s) = \arg \max_{i=1, \dots, C} ND_i \quad (16)$$

III. EXPERIMENT RESULT

A. Introduction to Datasets

In an artificial dataset, the set of 12 items $I = \{A, B, C, D, E, F, G, H, I, J, K, L\}$ are grouped into four parts where each part contains three items. For example, the first part contains A, B and C, the second part contains D, E, F, and so on. There are three class label ($C=3$) in the artificial dataset. The length of a sequence is randomly decided as a value from 3 to 20. Each class contains 200 sequences. For sequences with class label 1, 50% of items are be from group 1, 10% from group 2, 10% from group 3, and 30% from group 4. For sequences with class label 2, 50% of purchased items will be from group 2, 10% from group 1, 10% from group 3, and 30% from group 4. For sequences with class label 3, 50% of purchased items will be from group 3, 10% from group 1, 10% from group 2, and 30% from group 4.

In the first step of the proposed sequence classification method, the AprioriAll algorithm is applied to the artificial dataset and gets a set of sequential patterns. The support for AprioriAll algorithm is set to be 0.2 in this demonstration. In addition, the PSO-SA algorithm is applied to adjust the important weights of each sequential pattern. The initial parameter settings for the PSO-SA algorithm are shown in Table I.

TABLE I: THE PARAMETER SETTINGS FOR THE PSO-SA ALGORITHM

Parameter	Value	Remarks
w	1	weight of velocity in PSO
C_1	2	weight of the stochastic acceleration terms in PSO
C_2	2	weight of the stochastic acceleration terms in PSO
par	10	number of particles in PSO
t	300	number of iterations in PSO
h	20	examination range in PSO
α	0.99	adjustment factor of weight in PSO
β	0.95	adjustment factor of maximal velocity in PSO
T	100	initial temperature in SA
α'	0.99	factor of annealing in SA
m	30	operation steps in SA
n	10	operation steps in SA

B. The Result of the Proposed Classification Model

To show the benefit of the proposed classification model, another two classification methods, the non-binary model and the non-fuzzy model, are compared with the proposed classification method. The non-binary model is a classification strategy that predicts class labels according to the majority voting scheme. As the name suggested, the concept of pairwise coupling is not applied. In addition, weights of patterns are not used in the non-binary model. Non-fuzzy model is the strategy to predict class label based majority voting also. However, the pairwise coupling is applied to divide the dataset into $n(n-1)/2$ sub-database. However, the fuzzy preference relationship is not applied when combining the voting decision. In addition, weights of patterns are not applied to the non-fuzzy model.

Table II shows the comparison among three models using the training dataset. The classification accuracy of the proposed method (PSO-SA-BA classifier with fuzzy preference relation) is 93.16%. Clearly, the classification accuracy of this study is much better than another two classification models. The improvement between the proposed method and another two models is 28.66% and 29.66% respectively.

TABLE II: THE CLASSIFICATION ACCURACY ON TRAINING DATA

Classification method	Prediction accuracy
Non-binary model	64.5%
Non-fuzzy model	63.5%
Proposed model (PSO-SA-BA classifier with fuzzy preference relation)	93.16%

Moreover, 150 testing sequences are generated where for each class there are 50 sequences respectively. Table III shows the classification accuracy for the three methods on the testing data. Although the classification result on Table III is a little worse than the result on Table II, the proposed method is still superior to other two models.

TABLE III: THE CLASSIFICATION ACCURACY ON TESTING DATA

Classification method	Prediction accuracy
Non-binary model	64%
Non-fuzzy model	60.89%
Proposed model	84.44%

C. The Performance Comparison between the PSO Algorithm and the PSO-SA Algorithm

In this research, the PSO-SA algorithm is proposed to derive the optimal weights. To know the performance of the proposed PSO-SA algorithm for weight optimization, the traditional PSO algorithm is compared. Fig. 3 shows the classification accuracy and number of iterations for the PSO algorithm and the proposed PSO-SA algorithm respectively. The maximal classification accuracy for both PSO algorithm and PSO-SA algorithm can research 99.75%. However, the proposed PSO-SA algorithm can reach the maximal classification accuracy much earlier than PSO. The PSO-SA algorithm reaches the highest classification accuracy when the number of iterations is 116, while the PSO algorithm reaches the highest classification accuracy when the number of iterations is 448.

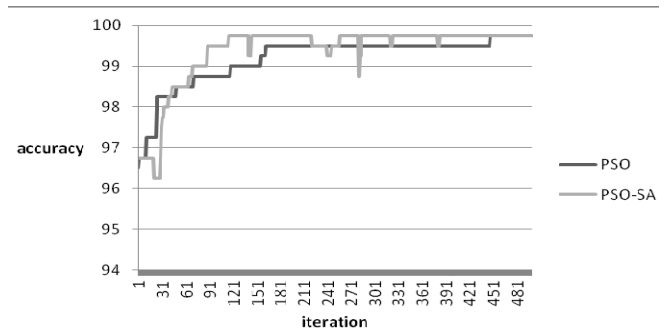


Fig. 3. The classification accuracy of adopting PSO and PSO-SA algorithms

The same experiment is conducted 30 times when different number of particles is tested. Table IV shows the number of iterations when the desired classification accuracy first reaches 98.5. It is clearly that the PSO algorithm spends much more time (iterations) to research optimal weights than the proposed PSO-SA algorithm does.

TABLE IV: THE NUMBER OF ITERATIONS WHEN ADOPTING PSO-SA AND PSO ALGORITHMS

Number of particles		PSO-SA	PSO
10	Min	16	23
	Max	359	978
	Avg	108.7	241.6
30	Min	14	5
	Max	263	714
	Avg	73.1	195.3
50	Min	3	3
	Max	250	608
	Avg	82.1	163.5

IV. CONCLUSION

However, previous studies seldom discuss how to obtain high classification accuracy when the data are multi-class sequences. To cope with it, this research proposes a classification method that integrates PSO-SA-BS classifier with fuzzy preference relations. The proposed classification model contains four steps. First, AprioriAll algorithm is used to generate sequential patterns for every class in the database respectively. Second, the pairwise coupling approach is applied to this study. The pairwise coupling approach generates $n(n-1)/2$ binary subsets, where n represents the total number of classes. Third, for each binary subset, a Particle Swarm Optimization with Simulated Annealing algorithm based Binary Sequence classifier, named PSO-SA-BS classifier, is constructed. In the PSO-SA-BS classifier, the similarity between two sequences is developed by a dynamic similarity computation method. Then, the PSO-SA optimization algorithm is developed to update the weights in the classifier so that the classification accuracy of each classifier can be maximized. Fourth, to aggregate the output from all PSO-SA-BS classifiers, the fuzzy preference relation between each pair of binary database is evaluated. According to the fuzzy preference relation, a class label with largest class score is assigned to the sequence.

Some future works can be done to improve this research. First, although similarity measure of sequences based on

string matching has been developed, it still can be modified further. For example, Euclidean distance, Minkowski distance, and other distance evaluation approaches can be taken into consideration. Another traditional classification method such as support vector machine (SVM), neural network (NN) and decision tree (DT) can be used to construct the binary sequence classifier in this research.

REFERENCES

- [1] K. Huang, H. Yang, I. King, and M. Lyu, "Learning classifiers from imbalanced data based on biased minimax probability machine," in *Proc. 4th IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004, pp. 558-563.
- [2] J. G. O. S. J. Joung and B. T. Zhang, "Protein sequence-based risk classification for human papillomaviruses," *Computers in Biology and Medicine*, vol. 36, pp. 353-667, 2006.
- [3] D. Bouchaffra and J. Tan, "Structural Hidden Markov Models Using a Relation of Equivalence: Application to Automotive Designs," *Data Mining and Knowledge Discovery*, vol. 12, no. 1, pp. 79-96, 2006.
- [4] A. D. Bruyn, J. C. Liechty, E. K. R. E. Huizingh, and G. L. Lilien, "Offering online recommendations with minimum customer input through conjoint-based decision aids," *Marketing Science*, vol. 27, no. 3, pp. 443-460, 2008.
- [5] N. Lesh, M. J. Zaki, and M. Ogihara, "Scalable feature mining for sequential data," *IEEE Intelligent Systems and their Applications*, vol. 15, pp. 48-56, 2000.
- [6] M. Kim and V. Pavlovic, "Sequence classification via large margin hidden Markov models," *Data mining and Knowledge Discovery*, vol. 23, pp. 322-344, 2010.
- [7] R. Agrawal and R. Srkant, "Mining sequential pattern," in *Proc. 11th International Conference on Data Mining*, 1995, pp. 3-14.
- [8] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," *The Annals of Statistics*, vol. 26, no. 2, pp. 451-471, 1998.
- [9] O. Arbell, G. M. Landau, and J. S. B. Mitchell, "Edit distance of run-length encoded strings," *Information Processing Letters*, vol. 83, pp. 307-314, 2002.
- [10] S. Soleymani, A. M. Ranjbar, S. Bagheri Shouraki, A. R. Shirani, and N. Sadati, "A new approach for bidding strategy of gencos using particle swarm optimization combined with simulated annealing method," *Iranian Journal of Science and Technology*, Transaction B, Engineering, vol. 31 (B3), pp. 303-315, 2007.
- [11] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. 4th IEEE International Conference on Neural Networks*, 1995, pp. 1942-1948.
- [12] R. C. Eberhart and Y. Shi, "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization," in *Proc. 11th Congress on Evolutionary Computation*, 2000, pp. 84-88.
- [13] Y. Shi and R. C. Eberhart, "Parameter Selection in Particle Swarm Optimization," *Lecture Notes in Computer Science*, vol. 1447, pp. 591-600, 1998.
- [14] P. C. Fourie and A. A. Groenwold, "The Particle Swarm Optimization Algorithm in Size and Shape Optimization," *Structural and Multidisciplinary Optimization*, vol. 23, no. 4, pp. 259-267, 2002.
- [15] T. Niknam, B. Amiri, J. Olamaei, and A. Arefi, "An efficient hybrid evolutionary optimization algorithm based on PSO and SA for clustering," *Journal of Zhejiang University Science A*, vol. 10, no. 4, pp. 512-519, 2009.
- [16] S. Orlovsky, "Decision-making with a fuzzy preference relation," *Fuzzy Sets and Systems*, vol. 1, pp. 155-167, 1978.



applications, and product data management.

Chieh-Yuan Tsai is a professor in the Department of Industrial Engineering and Management at Yuan-Ze University, Taiwan. He received his M.S. and Ph.D. degrees in Department of Industrial and Manufacturing Systems Engineering from the University of Missouri-Columbia, USA. His research activities include data mining, customer relationship management (CRM), RFID technology and



Chih-Jung Chen is a Ph.D. candidate in the Department of Industrial Engineering and Management at Yuan Ze University, Taiwan. He received his master degree at the same institute. His research activities include data mining, sequence classification, heuristic algorithms, product development process, and module product design.